NAME: NETID:

Exam 1.p.fa.2024

Principles of Computing

30th of October, 2024

UNIVERSITY OF NOTRE DAME

1 *Answer the following questions either True or False.*

The first five questions on this page are about the ***shell;*** the rest are about ***Python***.

i. The *root* of a UNIX-based file system is a directory named `/root`.

- ◯ *True*
- ◯ *False*

ii. The `cd` command can be used to change a directory's name.

- ◯ *True*
- ◯ *False*

iii. The name `~` is an alias for the current user's *"home"* directory.

- ◯ *True*
- ◯ *False*

iv. `mv(pumpkins.py, ~/patch)` will move `pumpkins.py` to the `patch` directory in the user's home folder.

- ◯ *True*
- ◯ *False*

v. `touch closet/skeletons.csv` will create a new file named `skeletons.csv` within `closet` in the home folder.

- ◯ *True*
- ◯ *False*

vi. `", ".join(["this is Halloween", "this is Halloween"])[19::1]` is valid syntax.

- ◯ *True*
- ◯ *False*

vii. Every value in *Python* is an object, and every object has a type.

- ◯ *True*
- ◯ *False*

viii. Every method is a function in *Python*.

- ◯ *True*
- ◯ *False*

ix. Every function returns a value.

- ◯ *True*
- ◯ *False*

x. `print("Hello, world!")` returns an object of type `str`.

- ◯ *True*
- ◯ *False*

NAME: NETID:

2 *Answer the following questions without justification.*

i. What is the value of `scary_string` at the end of the following block?

```
1 scary_string = "i am the one hiding under your bed".split(" ")
2 scary_string[0:2] += ["not"]
3 scary_string = " ".join(scary_string)
```

.....

ii. What is the result of the following expression?

```
1 len("".join("everyone hail to the pumpkin song\n".split(" ")))
```

.....

iii. What is the value of `spooky_func(" ")` with the definition below?

```
1 def spooky_func(ghost):
2   x  = "i am the shadow on the moon at night".split(ghost)
3   return x.insert(2, "not")
```

.....

iv. How many times will line 4 be evaluated in the block of code below?

```
1 def rattle(something):
2   return len(something) % 2 == 0
3 bones = list(range(101))
4 while rattle(bones):
5   bones = bones[:bones[-1]/2]
```

.....

v. What value is returned as a result of calling the function below?

```
1  def trick_or_treat():
2    ll = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
3    fear = 0
4    for x in ll:
5      for y, z in enumerate(x):
6        if z % (y + 1) == 0:
7          fear += z
8      if fear == 13:
9        return "trick"
10   return "treat"
```

.....

NAME: NETID:

3 *Answer the following questions without justification.*

i. What *type of error* will be thrown when this code is run?[1]

```
def jump_scare(days_left):
  return days_left == 7
upper_bound = input("Enter your maximum tolerance for terror: ")
for i in range(upper_bound):
  print(jump_scare(days_left))
```

[1] You may assume the user provides valid, sensible input (*e.g.*, 10) when prompted.

.....

ii. Indicate *all lines* in the following code block that would *throw an error.*

```
dungeon = []
for i in range(5):
  room = []
  for j in range(5):
    room += [i*j]
  dungeon.append(room)
if sum(dungeon) % 13 == 0:
  return "creepy"
return "wet"
```

.....

iii. The block of code below throws an error; on *what line* does it occur?

```
from random import randint
weight = "0"
giles_corey = "185"
while plea not in ["guilty", "not guilty"]:
  weight += int(f"{randint(1, 10)}")
  giles_corey -= int(weight)
  plea = "more weight"
```

.....

iv. What is `monster_sort([3, 1, -1, 0, 5, 3])` based on the code below?

```
def monster_sort(ll):
  rr = []
  for l in ll:
    scared = False
    for i, r in enumerate(rr):
      if (l < r) and (not scared):
        rr.insert(i, l)
        scared = True
    if not scared:
      rr.append(l)
  return rr
```

.....

NAME: NETID:

4 *Program a solution to the following question in Python.*

Given two vectors $x = [x_0, x_1, \ldots, x_{n-1}]$ and $y = [y_0, y_1, \ldots, y_{n-1}]$ of the same positive integer length n, we compute the *inner product* of x with y, which we denote $\langle x, y \rangle$, as follows.

$$\langle x, y \rangle = x_0 y_0 + x_1 y_1 + \cdots + x_{n-1} y_{n-1}$$

For example, the inner product of $[1, 2, 3]$ and $[2, 4, 8]$ is below.

$$\begin{aligned} \langle [1,2,3], [2,4,8] \rangle &= 1 \cdot 2 + 2 \cdot 4 + 3 \cdot 8 \\ &= 2 + 8 + 24 \\ &= 34 \end{aligned}$$

Write a function in *Python* that takes two lists of floating point numbers representing *vectors* as inputs and returns their *inner product* as output *if they are the same length,* but returns the string `"ERROR"` otherwise.

RESTRICTIONS:

- No use of `import` statements.
- No use of methods for any type.
- No use of the built-in functions:
  - `sorted()`
  - `sum()`
  - `max()`
  - `min()`
  - `abs()`
- No use of functions, methods, types, or control structures that we have not yet covered in class nor problem sets.

NOTE: you may name your function and your variables whatever you'd like.